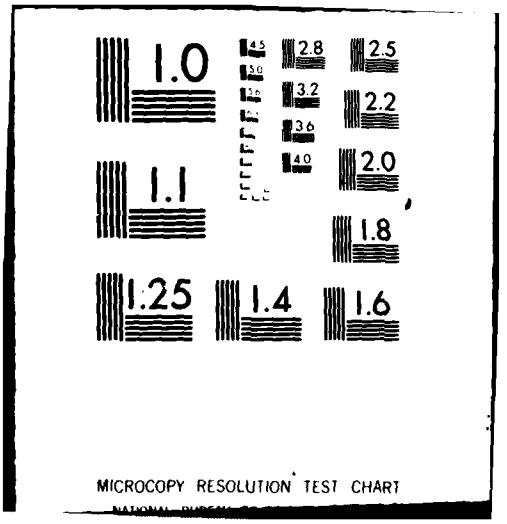
AD-A112 417 NAVAL RESEARCH LAB WASHINGTON DC F/G 9/2
A FORTRAN IV PLOT ROUTINE WITH HIDDEN LINE SUPPRESSION FOR USE --ETC(U)
MAR 82 H P COLEMAN
UNCLASSIFIED NRL-MR-4776 NL

END
DATE FILMED
4-82
DTIC


1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
4.5
5.0
MICROCOPY RESOLUTION TEST CHART

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER NRL Memorandum Report 4776	2. GOVT ACCESSION NO. AD-A112417	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) A FORTRAN IV PLOT ROUTINE WITH HIDDEN LINE SUPPRESSION FOR USE WITH SMALL COMPUTER SYSTEMS		5. TYPE OF REPORT & PERIOD COVERED Final report on this phase of NRL problem. 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) H. Paris Coleman		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Research Laboratory Washington, DC 20375		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 61153N; RR021-05-43; 53-0624-0
11. CONTROLLING OFFICE NAME AND ADDRESS Office of Naval Research Washington, DC 22217		12. REPORT DATE March 26, 1982 13. NUMBER OF PAGES 14
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Computer plotting
Three-dimensional plotting
Hidden line suppression

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A routine useful for plotting projections of surfaces representing a dependent variable as a function of two independent variables is described. This routine implements hidden line suppression. Simplification, to allow implementation on small computers, is emphasized. A Fortran IV listing of the routine is included.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

CONTENTS

INTRODUCTION 1

DESCRIPTION OF THE ROUTINE 1

DESCRIPTION OF PARAMETERS 2

SUMMARY. 3

REFERENCE 3

APPENDIX A - Listing of the plot routine 9

APPENDIX B - List of coordinate transformation routines. . 10

APPENDIX C - List of program used to generate plots . . . 11

A FORTRAN IV PLOT ROUTINE WITH HIDDEN LINE SUPPRESSION FOR USE WITH SMALL COMPUTER SYSTEMS

INTRODUCTION

A frequently encountered problem in computer graphics is the presentation of plots representing the value of a single dependent variable as a function of two independent variables. A commonly used solution to this problem consists of a depiction of a grid located on the surface representing the function. The curves comprising this grid are obtained by letting the remaining independent variable vary. The viewpoint is then established by coordinate rotation, with respect to the plane of the graphics medium, and the projection of the curves comprising the grid are plotted on this plane. In order to avoid a confused or ambiguous plot, some method must be used to suppress portions of the curves which would be hidden from the viewers position.

This report presents a routine which has been found useful in solving this hidden line problem. Simplification of the general problem has been used to provide a method which is suitable for use with small computer systems. The routine was developed for use with a Data General Corporation NOVA 800 with associated Tektronix Incorporated graphics terminal equipment. The coding, however, is Fortran IV and is easily adapted for use with other computer systems.

DESCRIPTION OF THE ROUTINE

Plots made using the routine are shown in figures 1 through 4. A listing of the Fortran IV source code is included as Appendix A. The comments included in the source listing should allow the program flow to be followed once the parameters and general method are explained.

Consider the function to be represented to be of the form: $y = F(x,z)$. We wish to depict curves of $y = F(x = \text{const.}, z)$ and $y = F(x, z = \text{const.})$. Taking one of these curves, the coordinates are transformed by a rotation of ϕ in the horizontal plane (x,z plane) and then tipped forward by an angle, θ. The projection in the plot coordinates is then obtained. The angle ϕ is required to be 45 degrees. This allows the entire problem to be handled in terms of indices of arrays, which simplifies the program and economizes on storage requirements. The coordinate transformations are the responsibility of the calling program. Listings of two routines, useful in this regard, are included in Appendix B.

The transformed data is stored in two arrays (IXPA and IYPA) giving the projected coordinates of the data. Hidden line suppression is accomplished by comparison with an array (IYMA) which stores the maximum Y encountered

Manuscript submitted January 25, 1982.

as a function of index. IYMA must be properly initialized in the calling program but is subsequently maintained by the plot routine. Plots should be accomplished from front to back. That is, constant z cuts must start with maximum z while constant x cuts must start with maximum x. Furthermore, at the completion of one set of variable equals a constant cut, IYMA must be properly reinitialized to the projected view of the front of the surface before proceeding to plot the other set.

Provision is made for updating of the IYMA array for values of the independent variable between plotted constant cuts. This gives a truer picture of the surface as a two dimensional solid (see figures 1 and 3) but may not be desired in some instances. For example, if either constant x or constant z cuts are used alone, the presentation obtained by not updating for intermediate surface may be preferred [1] (also see figure 4). For experimental data, intermediate values for the "surface" are frequently unknown and consequently demand the use of this option. The routine calculates the intersection of the line segment joining adjacent data points and the line segment joining the pertinent stored maximum y points and only plots the appropriate section of the functional curves. Two plot routines, MOVE and DRAW, are referenced in the routine. Since these are hardware dependent, listings are not included, but functional equivalents must be supplied by the user. MOVE (IX, IY) is equivalent to a "pen up" command while DRAW (IX, IY) is equivalent to "pen down", both with plotter motion to the point (IX, IY).

DESCRIPTION OF PARAMETERS

The call to the routine is:

```
CALL P3D (IXPA, IYPA, IYMA, MDIM, NDIM, NPTS, MLOW, IPL)
```

Arrays are dimensioned:

```
IXPA (0:NDIM), IYPA (0:NDIM), IYMA (MDIM:NDIM)
```

Definitions of parameters:

IXPA	is the x plot array.
IYPA	is the y plot array.
IYMA	is the array storing maximum value of y encountered for a particular index.
MDIM	should be set = -NDIM in the calling program
MLOW	must be set equal to the lowest value of index for IYMA (i.e., the left most point) in the calling program.
NPTS	is number of points to be plotted.
IPL =	0 don't plot, but update IYMA 1 plot and update IYMA

The relationship of the various indices is diagrammed in figure 5. This figure is a view looking down on the original coordinate system. Note that ϕ=45°, variation in θ does not effect these relationships. This is the key to the simplicity of the hidden line routine.

SUMMARY

A routine useful for plotting "three-dimensional" representations of single valued functions of two independent variables, or the equivalent in experimentally obtained data, has been described. The method used limits the rotation of point of view in order to economize on required computer resources, thus making the method applicable to small computer systems. Hidden line suppression is implemented, including the optional suppression of those portions of plotted cuts which would be hidden by the surface between cuts, in the event these intermediate values for the functions are known.

REFERENCE

[1] "A CDC 3800 Computer Subroutine for Producing Isometric Plots on the 565 Cal Comp Plotter", H.A. Brown, NRL Computer Note 31, 1 Aug 1969.

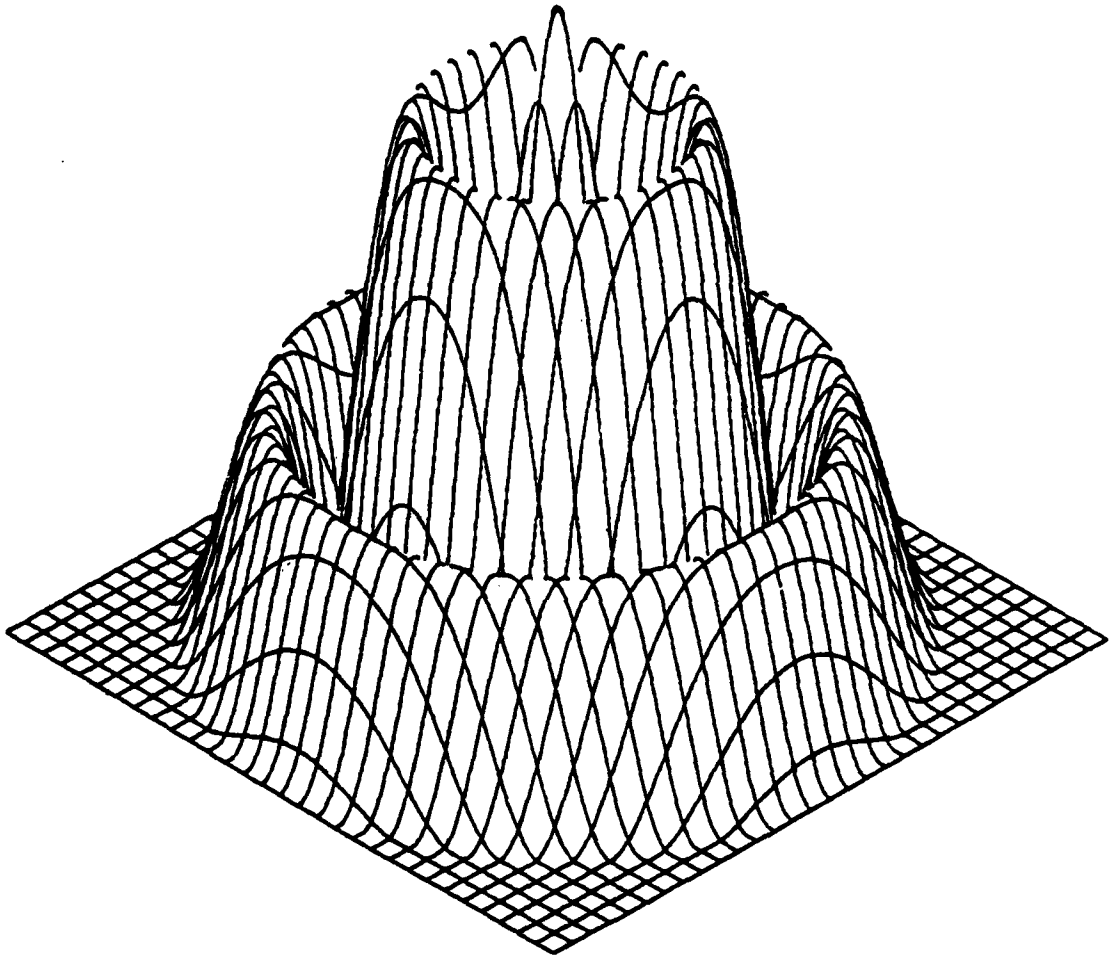


Fig. 1 - Plot of a surface using both constant x and constant z cuts. The YMAX array is updated between plotting cuts to obtain an accurate representation of the surface.

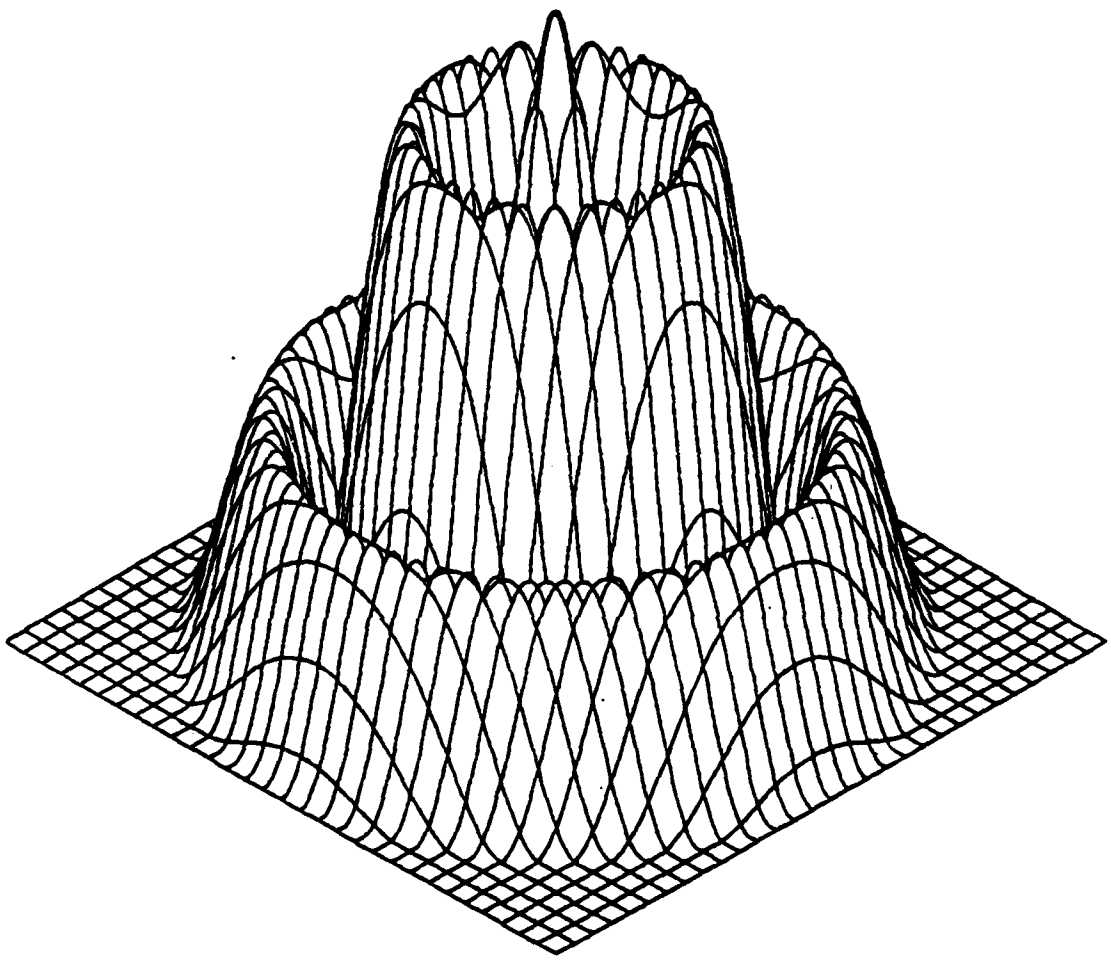


Fig. 2 - Plot of surface, as in figure 1, except the YMAX array is not updated between the plotted cuts. The deficiencies in the representation of the surface are apparent.

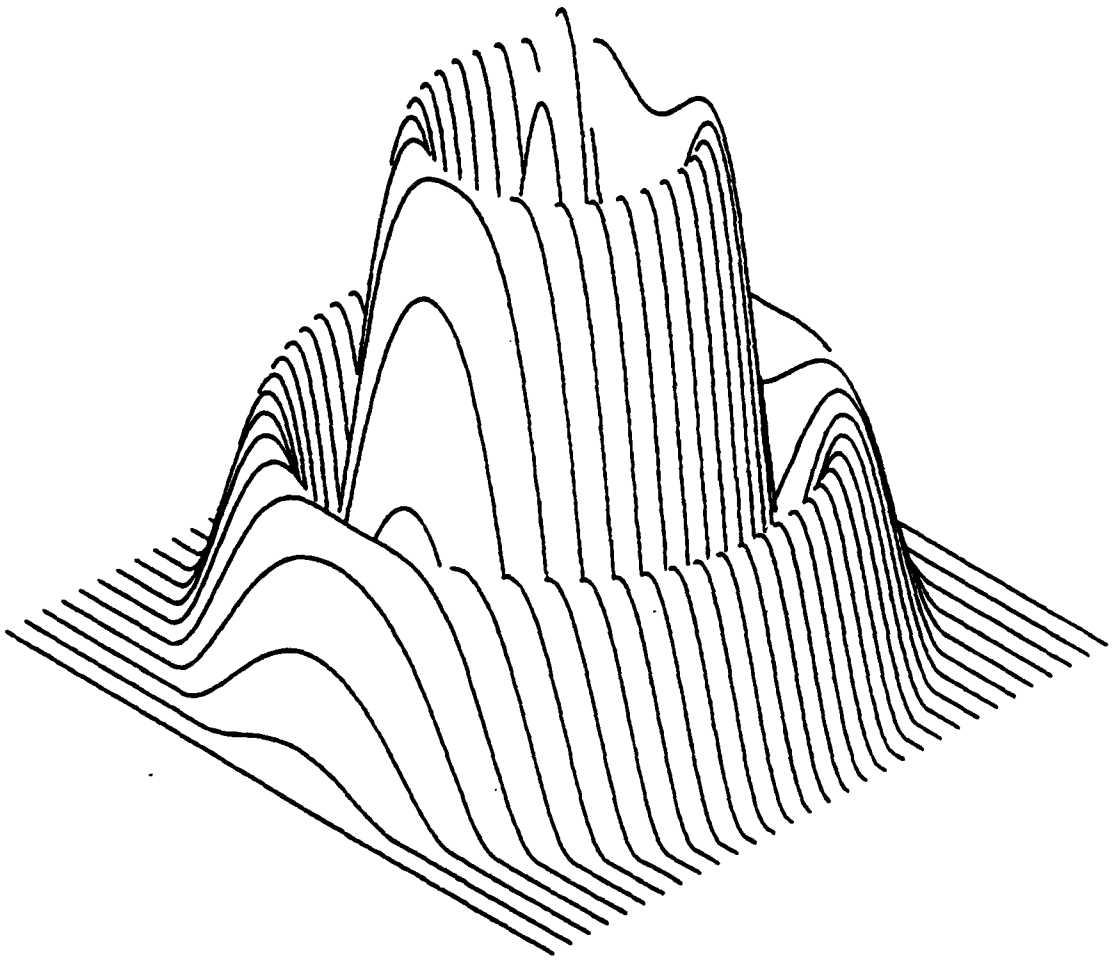


Fig. 3 - Plot of surface using constant x cuts alone. The YMAX array is updated between plotted cuts.

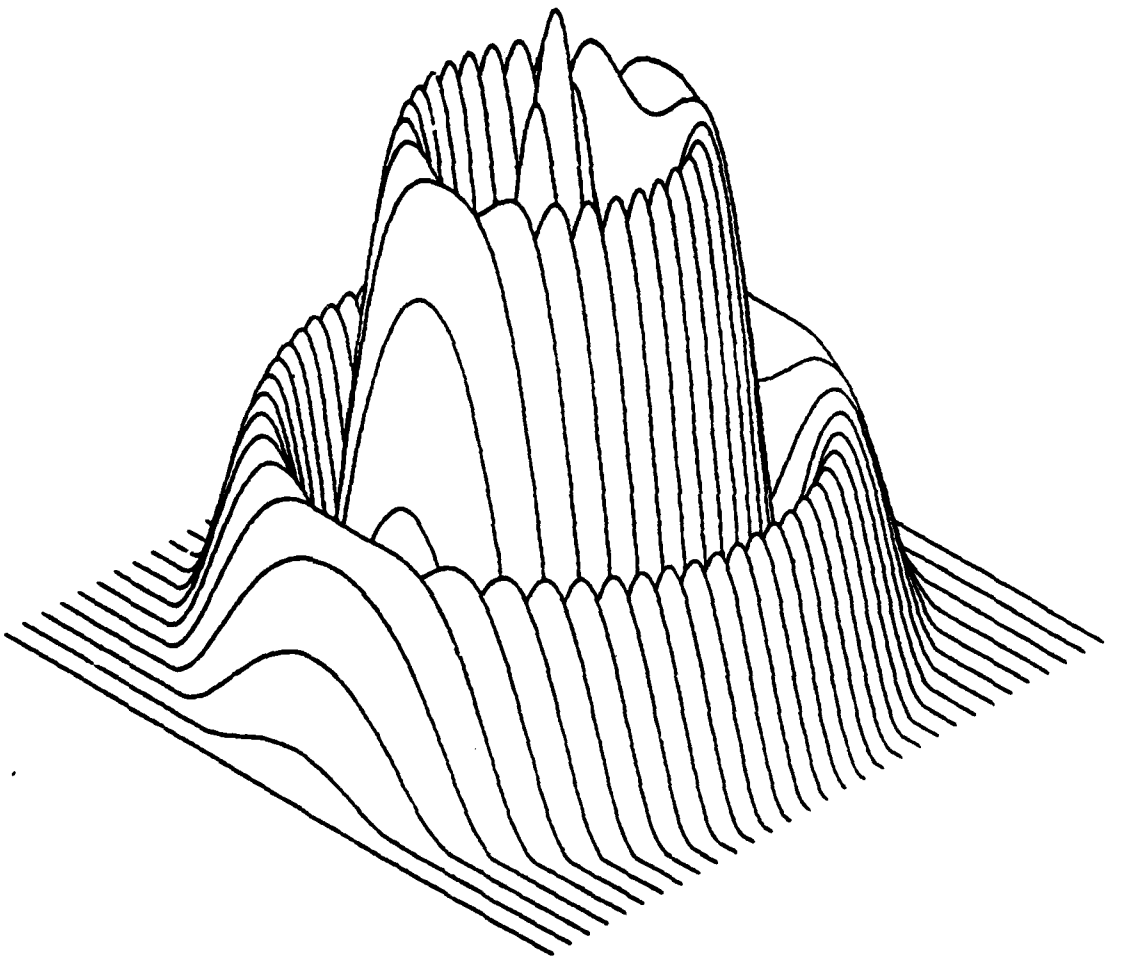


Fig. 4 - Plot of surface, as in figure 3, except the YMAX array is not updated between plotted cuts.

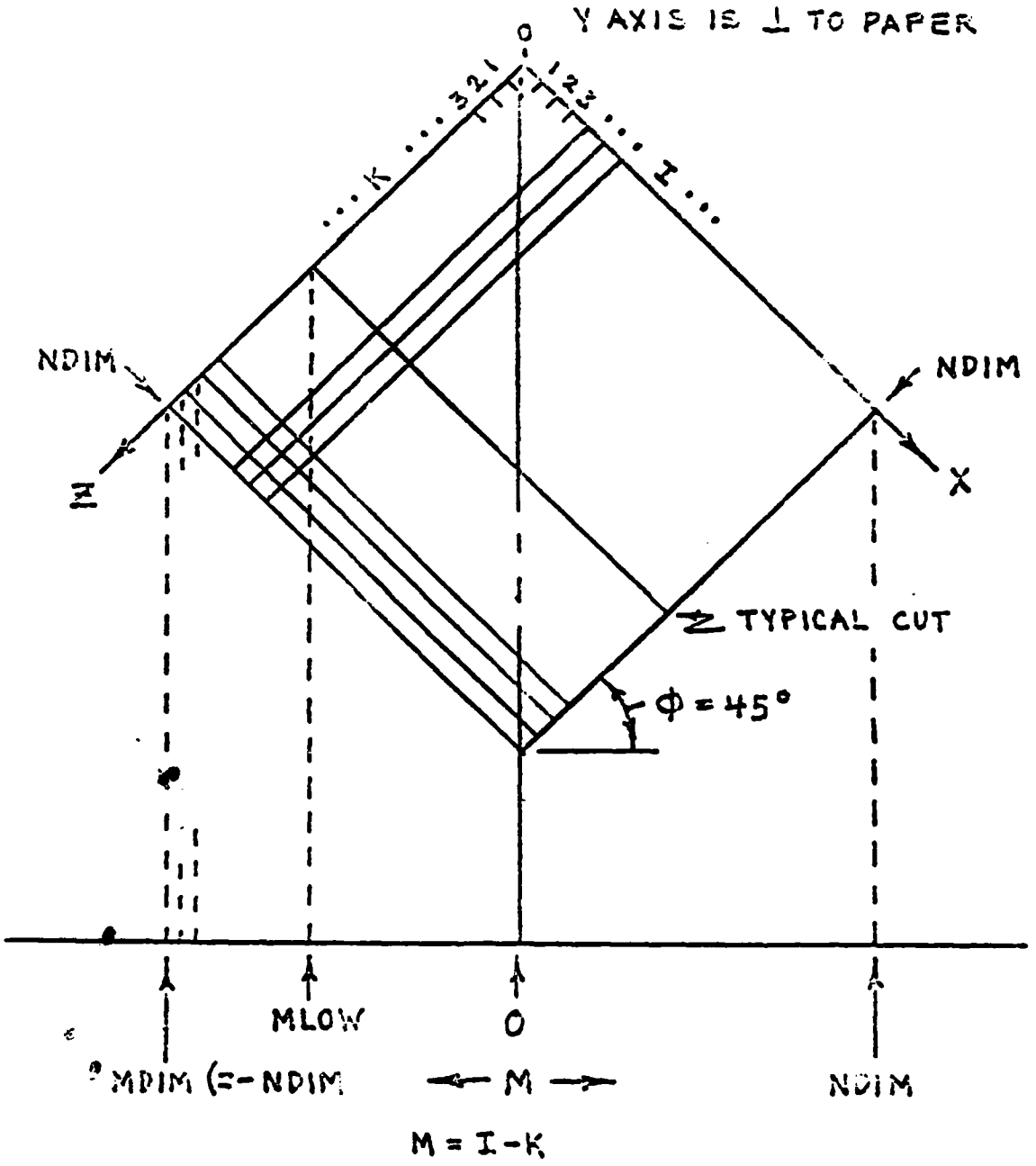


Fig. 5 - Diagram showing the relationship between indices of the various arrays

Appendix A — Listing of the plot routine

```
TYPE P3D.FR
      SUBROUTINE P3D(IXPA,IYPA,IYMA,MDIM,NDIM,NPTS,MLOW,IPL)
C     ROUTINE TO PLOT FROM ARRAYS BLANKING HIDDEN LINES
C     HPC 8/10/81
C     MODIFIED TO INTERPOLATE PARTIALLY HIDDEN VECTORS
C     HPC 9/1/81
      DIMENSION IXPA(0:NDIM),IYPA(0:NDIM),IYMA(MDIM:NDIM)
      IF(IPL.EQ.0) GO TO 41  ;DON'T PLOT, BUT UPDATE MAX ARRAY
      DO 31 IND=1,NPTS
      I=IND-1
      M=MLOW+I;INDEX OF MAX ARRAY
      IF(I.NE.0) GO TO 33
      CALL MOVE(IXPA(0),IYPA(0));POINT IS BEGINNING POINT
      GO TO 31
 33   IY1=IYPA(I-1)
      IY2=IYPA(I)
      IYM1=IYMA(M-1)
      IYM2=IYMA(M)
      IX1=IXPA(I-1)
      IX2=IXPA(I)
C  ARE BOTH PTS ABOVE (OR BELOW) YMAX?
      IF(.NOT.(IY1.LE.IYM1.AND.IY2.LE.IYM2)) GO TO 51
      CALL MOVE(IX2,IY2)   ;BOTH PTS BELOW
      GO TO 31
 51   IF(.NOT.(IY1.GT.IYM1.AND.IY2.GT.IYM2)) GO TO 52
      CALL DRAW(IX2,IY2)  ;BOTH PTS ABOVE
      GO TO 31
C  ONE END OF VECTOR IS HIDDEN, SET UP TO INTERPOLATE
 52   SM=FLOAT(IYM2-IYM1)/(IX2-IX1)  ;SLOPE OF YMAX
      SP=FLOAT(IY2-IY1)/(IX2-IX1)  ;SLOPE OF PLOT
      X=IX1+FLOAT(IY1-IYM1)/(SM-SP)  ;X AT PT OF INTERSECTION
      Y=SP*(X-FLOAT(IX1))+IY1  ;Y AT PT OF INTERSECTION
      IX=IFIX(X) ;INTEGER COORDINATES
      IY=IFIX(Y) ;OF INTERSECTION
C  DETERMINE WHICH END OF VECTOR IS HIDDEN AND PLOT
      IF(.NOT.(IY1.GT.IYM1.AND.IY2.LE.IYM2)) GO TO 53
      CALL DRAW(IX,IY)  ;LEFT ABOVE
      CALL MOVE(IX2,IY2);AND RIGHT BELOW
      GO TO 31
 53   CALL MOVE(IX,IY)  ;LEFT BELOW
      CALL DRAW(IX2,IY2);AND RIGHT ABOVE
 31   CONTINUE
 41   DO 35 IND=1,NPTS;UPDATE MAX ARRAY
      I=IND-1
      M=MLOW+I
 35   IF(IYPA(I).GT.IYMA(M)) IYMA(M)=IYPA(I)
      RETURN
      END
R
```

Appendix B — List of coordinate transformation routines useful in generating plots

```
TYPE ISOPR.FR
      SUBROUTINE ISOPR(X,Y,Z,XP,YP,IS)
C  HPC 9/9/81
C  PERFORM ISOMETRIC PROJECTION OF INPUT SYSTEM (X,Y,Z)
C  ON TO SCREEN COORDINATES (XP,YP)
C  IS=1 IF ZP IS POSITIVE, -1 IF NEGATIVE
C
      XP=(X-Z)*0.86603
      YP=Y-(X+Z)*0.5
      IS=1
      IF((X-Z).LT.0) IS=-1
      RETURN
      END
```

```
TYPE ROTC.FR
      SUBROUTINE ROTC(X,Y,Z,XP,YP,ZP)
C  HPC 9/9/81
C  ROTATION OF COORDINATE SYSTEM (X,Y,Z) TO NEW
C  COORDINATE SYSTEM (XP,YP,ZP)
C  ROTATION ABOUT X (TOWARD Z) BY THETA
C  THEN ROTATE ABOUT NEW Y (TOWARD ORIGINAL Z)
C  BY PHI
C  CTH,STH ARE SINE,COSINE OF THETA
C  CHP,SPH ARE SINE,COSINE OF PHI
C  THESE MUST BE SET IN COMMON
C  FOR EXAMPLE,BY CALL TO SROTC
C
      COMMON/CROTC/CPH,SPH,CTH,STH
      XP=X*CPH-Z*SPH
      YP=Y*CTH-Z*CPH*STH-X*SPH*STH
      ZP=Z*CPH*CTH+X*SPH*CTH+Y*STH
      RETURN
      END
```

Appendix C — Listing of program used to generate plots shown in the figures

```
C DEMO PROGRAM FOR HIDDEN LINE PLOTTING, MPC 9/9/81
      DIMENSION IYPA(0:150),IXPA(0:150),IYMA(-150:150)
      CALL INITT
      IS=0
      PI=3.1415927
      RAD=180./PI
 99   ACCEPT* IXC? ',IXC  ;CENTER OF SCREEN COORDINATES
      ACCEPT* IYC? ',IYC  ;'
      ACCEPT* NMAX? ',NMAX  ;MAX LENGTH OF AXES IN SCREEN POINTS
      FNMAX=NMAX  ;FLOAT AXIS LENGTH
      NDIV=150  ;NUMBER OF POSSIBLE DIVISION ALONG AXES
      NPTS=NDIV+1  ;NUMBER OF PLOT POINTS
      ACCEPT* NCUT? ',NCUT  ;NUMBER OF DIVISIONS ON AXES BETWEEN PLOTTED GRIDS
      ACCEPT* UPDATE MAX ARRAY BETWEEN GRIDS? (TYPE 1 FOR YES) ',NS
      MDIM=-150  ;SET LOWER DIMENSION OF YMAX ARRAY
      NDIM=150  ;SET UPPER DIMENSIONS OF ARRAYS
      TYPE
      TYPE* AT BREAKS WITH SOUND OF BELL*
      TYPE* RESUME OPERATION BY INPUTTING NUMBER AND CR*
      TYPE* (INPUT OF 1 RESTARTS PROGRAM AT BEGINNING)*
      TYPE
      ACCEPT* INPUT 1 FOR PLOTTER OUTPUT ',NP  ;SELECT SCREEN OR PLOTTER
      IF(NP.NE.1) GO TO 25
      TYPE* SWITCH TO AUX OUTPUT, BACK TO TTY AT END OF PLOT*
      PAUSE
      CALL PLON
 25   IF(NP.NE.1) CALL CLEAR  ;CLEAR SCREEN FOR GRAPHICS
C
C  GO TO PLOTS
C
      NLIM=2*NDIV+1
      DO 29 N=1,NLIM  ;CLEAR MAX ARRAY
      I=N-NDIV-1
 29   IYMA(I)=0
      CALL GRAPH(96)  ;SET GRAPH MODE AND VECTOR TYPE
C
C  DO Z=CONST CUTS
C
      DO 37 M=1,NPTS  ;LOOP OVER Z
      J=M-1
      Z=FNMAX-J*(FNMAX/NDIV)  ;Z COORDINATE OF CUT
      MLOW=MDIM+J  ;SET LEFT MOST POINT OF MAX ARRAY
      IF(MOD(J,NCUT).EQ.0) GO TO 34  ;DETERMINE WHETHER TO PLOT
      IPL=0
      GO TO 36
 34   IPL=1
 36   IF(NS.EQ.0.AND.IPL.EQ.0) GO TO 37 ;DO NOT UPDATE FOR INTERMEDIATES
C
      DO 33 N=1,NPTS  ;PUT TRANSFORMED X,Y IN ARRAYS AND PLOT
      I=N-1
      X=I*(FNMAX/NDIV)
      TX=X-FNMAX/2.  ;TRANSLATE
      TZ=Z-FNMAX/2.
      RHO=SQRT(TX**2+TZ**2)
      ARG=(RHO/(FNMAX/2.))*5.*PI/2.
      Y=FNMAX*ABS(COS(ARG))*COS(RHO*PI/FNMAX)
      IF(RHO.GT.(FNMAX/2.)) Y=0.
```

Appendix C (Cont'd) — Listing of program used to generate plots shown in the figures

```
      CALL ISOPR(X,Y,Z,XP,YP,IS)  ;ISOMETRIC PROJECTION (FLOATING)
      IXPA(I)=XP+IXC  ;TRANSLATE TO SCREEN COORDINATES
 33   IYPA(I)=YP+IYC
      CALL P3D(IXPA,IYPA,IYMA,MDIM,NDIM,NPTS,MLOW,IPL)  ;PLOT ARRAY
 37   CONTINUE
      CALL MOVE(0,10)  ;MOVE NEAR BOTTOM LEFT
      CALL ALPHA(3)  ;RETURN TO ALPHA MODE
      IF(NP.EQ.1) CALL PLOFF
C
      CALL BELL
      ACCEPT ICON
      IF(ICON.EQ.1) GO TO 99  ;RESTART
      IF(NP.EQ.1) CALL PLON
      CALL GRAPH(96)  ;RESET GRAPH MODE AND VECTOR TYPE
C
C  DO X=CONST CUTS
C
      DO 39 M=1,MLIM  ;RECLEAR MAX ARRAY
      I=M-MDIV-1
 39   IYMA(I)=0
C
      DO 47 M=1,NPTS  ;LOOP OVER X
      J=M-1
      X=FNMAX-J*(FNMAX/NDIV)  ;X COORDINATE OF CUT
      MLOW=-J  ;LEFTMOST POINT OF MAX ARRAY
      IF(MOD(J,NCUT).EQ.0) GO TO 44  ;PLOT OR UPDATE?
      IPL=0
      GO TO 46
 44   IPL=1
 46   IF(MS.EQ.0.AND.IPL.EQ.0) GO TO 47  ;DO NOT UPDATE FOR INTERMEDIATES
      DO 43 M=1,NPTS  ;PUT TRANSFORMED COORDINATES IN ARRAYS AND PLOT
      I=M-1
      Z=FNMAX-I*(FNMAX/NDIV)
      TX=X-FNMAX/2.
      TZ=Z-FNMAX/2.
      RHO=SQRT(TX**2+TZ**2)
      ARG=(RHO/(FNMAX/2.))*.5*PI/2.
      Y=FNMAX*ABS(COS(ARG))*COS(RHO*PI/FNMAX)
      IF(RHO.GT.(FNMAX/2.)) Y=0.
      CALL ISOPR(X,Y,Z,XP,YP,IS)
      IXPA(I)=XP+IXC
 43   IYPA(I)=YP+IYC
      CALL P3D(IXPA,IYPA,IYMA,MDIM,NDIM,NPTS,MLOW,IPL)
 47   CONTINUE
      CALL MOVE(0,0)  ;MOVE TO LOWER LEFT
      IF(NP.EQ.1) CALL PLOFF
      CALL ALPHA(3)  ;RETURN TO ALPHA MODE
      CALL BELL  ;SIGNAL END OF PLOT
      ACCEPT ICON
      IF(ICON.EQ.1) GO TO 99  ;RESTART
      END
```